Lecture 9

19 FEB 2014

start w/ basic quantum algorithms & phase kickback trick commonly employed in them

Consider the action of a CNOT gate when target is prepared in $|-\rangle$ state

$$|0\rangle|-\rangle \xrightarrow{\text{CNOT}} |0\rangle|-\rangle = |0\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$$

but

$$|1\rangle|-\rangle = |1\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right) = \frac{|10\rangle-|11\rangle}{\sqrt{2}}$$

$$\xrightarrow{\text{CNOT}} \frac{|11\rangle-|10\rangle}{\sqrt{2}} = |1\rangle\left(\frac{|1\rangle-|0\rangle}{\sqrt{2}}\right)$$

$$= (-1)|1\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$$

$$= (-1)|1\rangle|-\rangle$$

so we can neatly summarize this as

$$|b\rangle|-\rangle \rightarrow (-1)^b|b\rangle|-\rangle$$

where $b \in \{0,1\}$

Now, we showed before how to promote any ~~reversible~~ Boolean circuit computing $f: \{0,1\}^n \to \{0,1\}$ to a reversible circuit, which we can then implement to have the following action U_f on the comp. basis

$x \in \{0,1\}^n \quad y \in \{0,1\}$

$$|x,y\rangle \xrightarrow{U_f} |x, y \oplus f(x)\rangle$$

Now suppose as above that we prepare "target" register in $|-\rangle$ state, then

$$|x\rangle|-\rangle = |x,0\rangle - |x,1\rangle$$

$$\to |x, f(x)\rangle - |x, f(x)\oplus 1\rangle$$

Suppose that $f(x)=0$. Then the above is equal to

$$|x,0\rangle - |x,1\rangle = |x\rangle|-\rangle$$

If $f(x)=1$, then it is equal to

$$|x,1\rangle - |x,0\rangle = |x\rangle(-1)|-\rangle = (-1)|x\rangle|-\rangle$$

So we can write the action of the circuit as

$$|x\rangle|-\rangle \rightarrow (-1)^{f(x)}|x\rangle|-\rangle$$

This effect is called the "phase kickback" trick, b/c this particular input for target ($|-\rangle$) causes a phase to "kickback" the source register depending on the value of the function.

So, now suppose that we prepare the initial register in a superposition of all possible states (To do so, apply $H^{\otimes n}$

$$|0\rangle|-\rangle \rightarrow \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle|-\rangle$$

$$\xrightarrow{U_f} \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} (-1)^{f(x)}|x\rangle|-\rangle$$

So we can prepare a rather complicated quantum state... now show how it can

Deutsch-Jozsa Algorithm

(first alg. to show an exponential separation between quantum & deterministic classical comp.)

In this problem, We are given a promise that a function f is either constant

(i.e., $f(x) = y$ same $\forall x$)

or balanced

(i.e., $f(x) = 0$ for exactly half of the inputs & $f(x) = 1$ for the other half)

(we are also given "black box" access to function)

the problem is to decide which is the case.

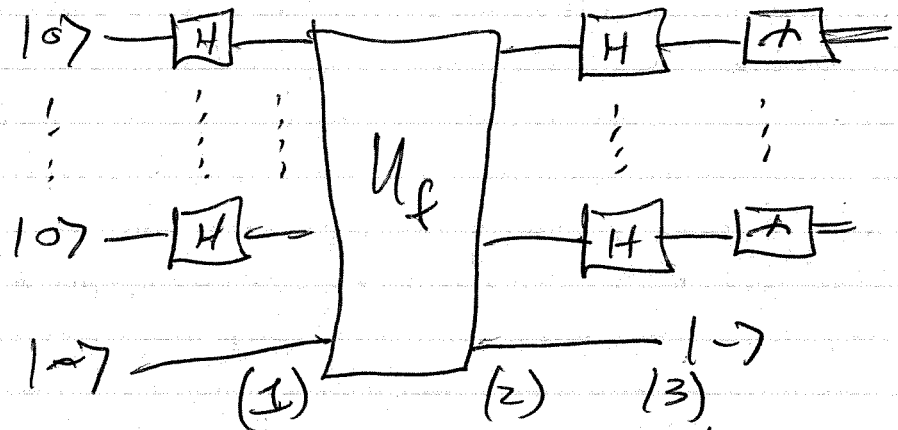
1st think about a classical algorithm to solve the problem

Suppose you have already queried half of the inputs (2^{n-1} values). At this point, in the worst case, you still might not know whether the function is constant or balanced. So you would need to make one more query to decide, for a total of $2^{n-1}+1$ queries

We will take black box access to mean a unitary that we can query in superposition, i.e.,

$$|x\rangle|y\rangle \xrightarrow{U_f} |x\rangle|y \oplus f(x)\rangle$$

circuit for the algorithm is



Let's track evolution through the circuit:

$$|0\rangle^{\otimes n}|-\rangle \xrightarrow{(1)} \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle|-\rangle$$

$$\xrightarrow{(2)} \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} (-1)^{f(x)} |x\rangle|-\rangle$$

Effect of Hadamard on

$$H|0\rangle \to |+\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}$$

$$H|1\rangle \to |-\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

abbreviate as

$$H|b\rangle \to \frac{|0\rangle + (-1)^b|1\rangle}{\sqrt{2}} = \frac{1}{\sqrt{2}} \sum_{c \in \{0,1\}} (-1)^{bc}|c\rangle$$

for an n-bit string x, the action will be

$$H^{\otimes n}|x_1 \cdots x_n\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + (-1)^{x_1}|1\rangle\right) \cdots \frac{1}{\sqrt{2}}\left(|0\rangle + (-1)^{x_n}|1\rangle\right)$$

$$= \frac{1}{\sqrt{2^n}} \sum_{z_1,\ldots,z_n \in \{0,1\}} (-1)^{\sum_{i=1}^{n} x_i z_i} |z_1, \ldots, z_n\rangle$$

abbreviate as

$$H^{\otimes n}|x\rangle = \frac{1}{\sqrt{2^n}} \sum_{z \in \{0,1\}^n} (-1)^{x \cdot z} |z\rangle$$

So the effect of the last array of Hadamards is to

$$\rightarrow \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} (-1)^{f(x)} |x\rangle \left(\frac{1}{\sqrt{2^n}} \left(\sum_{z=0}^{2^n-1} (-1)^{x \cdot z} |z\rangle \right) \right) \rightarrow$$

Now finally a computational basis measurement is performed on 1st register

What is the amplitude for all zeros state?

calculate

$$\frac{1}{2^n} \langle 0|^{\otimes n} \sum_{x=0}^{2^n-1} (-1)^{f(x)} \sum_{z=0}^{2^n-1} (-1)^{x \cdot z} |z\rangle$$

$$= \frac{1}{2^n} \sum_{x,z=0}^{2^n-1} (-1)^{f(x)} (-1)^{x \cdot z} \langle 0|z\rangle$$

$$= \frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{f(x)}$$

Suppose f is constant $\Rightarrow$ then the amplitude is either $+1$ or -1

So probability of measuring $|0\rangle^{\otimes n}$ is 1

for this case

Suppose f is balanced.

Then there are an equal number of zeros + ones for $f(x)$ so that

$$\frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{f(x)} = 0$$

+ we are certain to measure some other state besides $|0\rangle^{\otimes n}$

So thus is how we decide w/ just one query (although really kind of like 2 queries since we uncomputed)

Note that there is a classical probabilistic algorithm that can decide this efficiently using 2 queries w/ probability $\geq 2/3$ can amplify to be $\geq 1 - 2^{-n}$ w/ $O(n)$ queries

Simon's algorithm

Consider a function $f: \{0,1\}^n \to X \subseteq \{0,1\}^n$ where there is a promise that $\exists$ hidden string $s = s_1 \ldots s_n$ such that $f(x) = f(y)$ iff $x = y$ or $x = y \oplus s$

The goal is to determine the hidden string.

Before describing the algorithm, consider that

$$H^{\otimes n}|x\rangle = \frac{1}{\sqrt{2^n}} \sum_{z \in \{0,1\}^n} (-1)^{x \cdot z} |z\rangle$$

So if we apply $H^{\otimes n}$ to $|0\rangle + |s\rangle$, then

$$H^{\otimes n}\left(\frac{1}{\sqrt{2}}(|0\rangle + |s\rangle)\right) = \frac{1}{\sqrt{2^{n+1}}} \sum_z |z\rangle + \frac{1}{\sqrt{2^{n+1}}} \sum_z (-1)^{s \cdot z} |z\rangle$$

$$= \frac{1}{\sqrt{2^{n+1}}} \sum_z \left(1 + (-1)^{s \cdot z}\right) |z\rangle$$

Observe that if $s \cdot z = 1$, then

$1 + (-1)^{s \cdot z} = 0$ & $|z\rangle$ vanishes in the superposition

If $s \cdot z = 0$, then $|z\rangle$ remains w/ amplitude $\frac{1}{\sqrt{2^{n-1}}}$

Define the "dual space" of s to be $s^{\perp} = \{ z \mid s \cdot z = 0 \}$

$s^{\perp}$ is a vector subspace of $\mathbb{F}_2^n$ orthogonal to subspace $\{0, s\} = S$, known as orth. complement of S (denote as $S^{\perp}$)

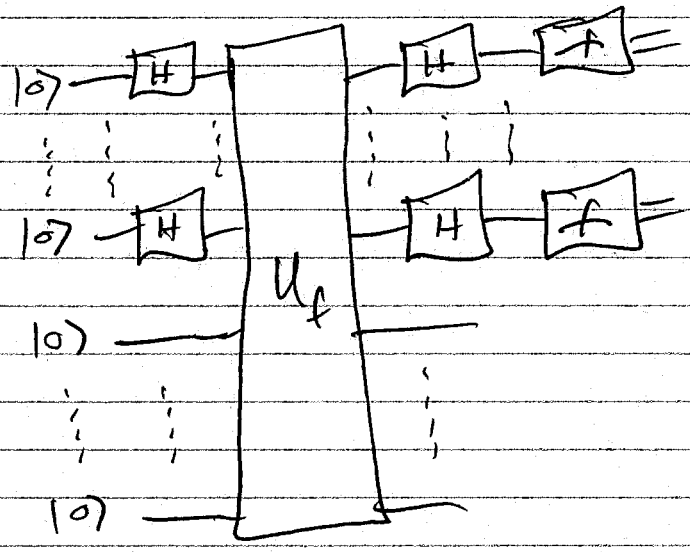
$\Rightarrow \dim(S) + \dim(S^{\perp}) = n$

so, $\dim(S^{\perp}) = n-1$

So, we can write

$$H^{\otimes n}\left(\frac{1}{\sqrt{2}}(|0\rangle + |s\rangle)\right) = \frac{1}{\sqrt{2^{n-1}}} \sum_{z \in s^{\perp}} |z\rangle$$

Simon's algorithm



can evaluate U_f as

$$U_f |x\rangle|b\rangle = |x\rangle|b \oplus f(x)\rangle$$

So, we analyze:

$$|0\rangle^{\otimes n}|0\rangle^{\otimes m} \rightarrow \frac{1}{\sqrt{2^n}} \sum_x |x\rangle|0\rangle$$

$$\xrightarrow{U_f} \frac{1}{\sqrt{2^n}} \sum_x |x\rangle|f(x)\rangle$$

we can partition all strings $\{0,1\}^n$ into 2^{n-1} pairs of strings of the form $\{x, x \oplus s\}$. Let I be a subset of $\{0,1\}^n$ consisting of one representative of each set.

Then we can write the state as

$$\frac{1}{\sqrt{2^{n-1}}} \sum_{x \in I} \frac{1}{\sqrt{2}}\left(|x\rangle + |x \oplus s\rangle\right)|f(x)\rangle$$

~~Assuming that $HX = ZH$~~

~~We~~ Measure 2nd register, then 1st register will collapse to

$$\frac{1}{\sqrt{2}}\left(|x\rangle + |x \oplus s\rangle\right)$$

Now use prior result & the fact that $HX = ZH$ to conclude that

$$H^{\otimes n}\left(\frac{1}{\sqrt{2}}\left(|x\rangle + |x \oplus s\rangle\right)\right) = \frac{1}{\sqrt{2^{n-1}}} \sum_{z \in S^{\perp}} (-1)^{x \cdot z} |z\rangle$$

Now measuring this register gives a member of $S^{\perp}$ selected randomly @ random (call it w_i)

Keep doing this until we sample enough vectors $\{w_i\}$ to span the dual space.

Then solve the equation

$$[W][s] = [0]$$

using Gaussian elimination to find the value of s.

expected # of queries is

$\approx n$

Best classical algorithm requires exponential time.